

Article

Detecting Pipeline Pathways in Landsat 5 Satellite Images with Deep Learning

Jan Dasenbrock , Adam Pluta, Matthias Zech and Wided Medjroubi

DLR Institute of Networked Energy Systems, 26129 Oldenburg, Germany; adam.pluta@dlr.de (A.P.); matthias.zech@dlr.de (M.Z.); wided.medjroubi@dlr.de (W.M.)

* Correspondence: jan.dasenbrock@dlr.de

Abstract: Energy system modeling is essential in analyzing present and future system configurations motivated by the energy transition. Energy models need various input data sets at different scales, including detailed information about energy generation and transport infrastructure. However, accessing such data sets is not straightforward and often restricted, especially for energy infrastructure data. We present a detection model for the automatic recognition of pipeline pathways using a Convolutional Neural Network (CNN) to address this lack of energy infrastructure data sets. The model was trained with historical low-resolution satellite images of the construction phase of British gas transport pipelines, made with the Landsat 5 Thematic Mapper instrument. The satellite images have been automatically labeled with the help of high-resolution pipeline route data provided by the respective Transmission System Operator (TSO). We have used data augmentation on the training data and trained our model with four different initial learning rates. The models trained with the different learning rates have been validated with 5-fold cross-validation using the Intersection over Union (IoU) metric. We show that our model can reliably identify pipeline pathways despite the comparably low resolution of the used satellite images. Further, we have successfully tested the model's capability in other geographic regions by deploying satellite images of the NEL pipeline in Northern Germany.

Keywords: pipeline detection; CNN; Landsat 5; U-Net; gas transport network



Citation: Dasenbrock, J.; Pluta, A.; Zech, M.; Medjroubi, W. Detecting Pipeline Pathways in Landsat 5 Satellite Images With Deep Learning. *Energies* **2021**, *14*, 5642. <https://doi.org/10.3390/en14185642>

Academic Editor: Gian Giuseppe Soma

Received: 30 June 2021

Accepted: 3 September 2021

Published: 8 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Integrating renewable energy sources (RES) into the existing energy infrastructure is an important research topic in energy system modeling. Recent studies investigate the synergies offered by sector-coupling technologies [1] and how they can contribute to the integration of higher shares of RES in the energy system. Especially, the coupling between gas and power grids [2,3] is of major interest, where, for example, modern power-to-gas (P2G) technology could be used to store electricity in the gas network [4]. With the rising importance of the different energy grids, energy system modelers face the central problem of insufficient data. Transmission Systems Operators (TSOs) of transport grids are generally reserved when it comes to publishing their infrastructural data sets. Thus, a lot of effort has been undertaken to create such open-source data for energy networks.

The *OpenStreetMap* project has already proven to be a legitimate source for data mining projects regarding the European power grid, e.g., [5,6]. However, equal efforts for collecting gas transport pipelines in *OpenStreetMap* currently seem less promising, as the respective data is still very sparse [7]. Since gas transport pipelines are buried underground and thus are not identifiable with the naked eye, they are hard to map without additional efforts. Therefore, it comes without surprise that current open-source data models of the gas transport grid [8,9] have issues, limiting the available level of detail and geographic coverage.

This work shows that modern remote sensing using satellite images is a well-suited option to gather infrastructural data of the gas transport network with very high geographic

accuracy. Even though remote sensing has already been frequently applied to pipelines, such applications leaned more towards monitoring purposes like leakage detection [10] or environmental change detection [11]. For such undertakings, the underlying pipeline network data is usually provided by the respective pipeline operators. From a machine learning standpoint, detecting pipeline routes is an entirely different challenge that requires the segmentation of line-like infrastructure objects in images. Recently, Convolutional Neural Networks (CNN) were successful in edge-detection [12] and in pixel-wise line detection in a fully convolutional setting [13].

Our paper is structured as follows: In Section 2, we describe our methodology in detail before the training and validation results are documented in Section 3. Conclusions are drawn in Section 4. The paper is based in part on the work done for this master's thesis [14].

2. Methods

In the following, we describe our methodology, which includes the model architecture of our deep learning network, the extraction of suitable images and training data, the satellite image preparation, and the training and validation process. Finally, we also show the results of applying our model to a new geographic region.

2.1. Model

Deep learning describes a set of algorithms that allows computational models to learn data representations with multiple levels of abstraction [15]. Its applications range from the improvement of industrial processes [16,17] to the support in medical diagnostics [18] to remote sensing [19,20] and computer vision [21]. In computer vision, problems like the detection of pipeline-like objects can be solved by classifying each pixel of an image to a respective object class, a process known as *semantic image segmentation* [22]. Using Convolutional Neural Networks (CNN) has shown success in addressing the challenges posed by semantic image segmentation [23,24]. One particular CNN architecture is U-Net, initially developed for multi-class image segmentation of biomedical images with little training data [25]. The U-Net has gained a lot of popularity in recent years [26–31], mainly due to its success in benchmark tests as well as its simplicity, efficiency, and robustness. Furthermore, the U-Net achieves a pixel-wise segmentation through the model architecture avoiding computational expensive and less accurate sliding-window approaches for localizing the target classes in the images [25].

A schematic illustration of the model used in this work can be seen in Figure 1. The architecture takes 64×64 px images with five channels as input and yields a map of the same dimensions with one channel. It contains a total number of 23 convolutional layers and can be divided into 2 parts: a contracting part (left) and an expansive part (right). The left part is the downsampling part, also called *the encoder*. It is a convolutional network for the increase of feature information and the reduction of spatial information. This is achieved by consecutive sets of two 3×3 convolutions, followed by a ReLU layer and a 2×2 max-pooling operation. Therefore, the encoder reduces the spatial resolution from 64×64 px to 2×2 px. At the same time, the number of feature layers increases from 5 to 512. The right part is the up-sampling part, also called *the decoder*. Its goal is to combine spatial and feature information through consecutive sets of two 3×3 convolutions with intermediate batch normalization layers, followed by a 2×2 up-sampling layer. The decoder reduces the number of feature layers from 512 to 1 while increasing the spatial resolution back to 64×64 px. Additionally, skip connections perform a concatenation between distant layers. Implemented symmetrically between encoder and decoder, skip connections help to share fine-grained feature information to reconstruct the image from the learned features better.

The model was implemented using the Python library *Segmentation Models* [32] which is based on the machine learning and neural network libraries *Tensorflow* and *Keras*.

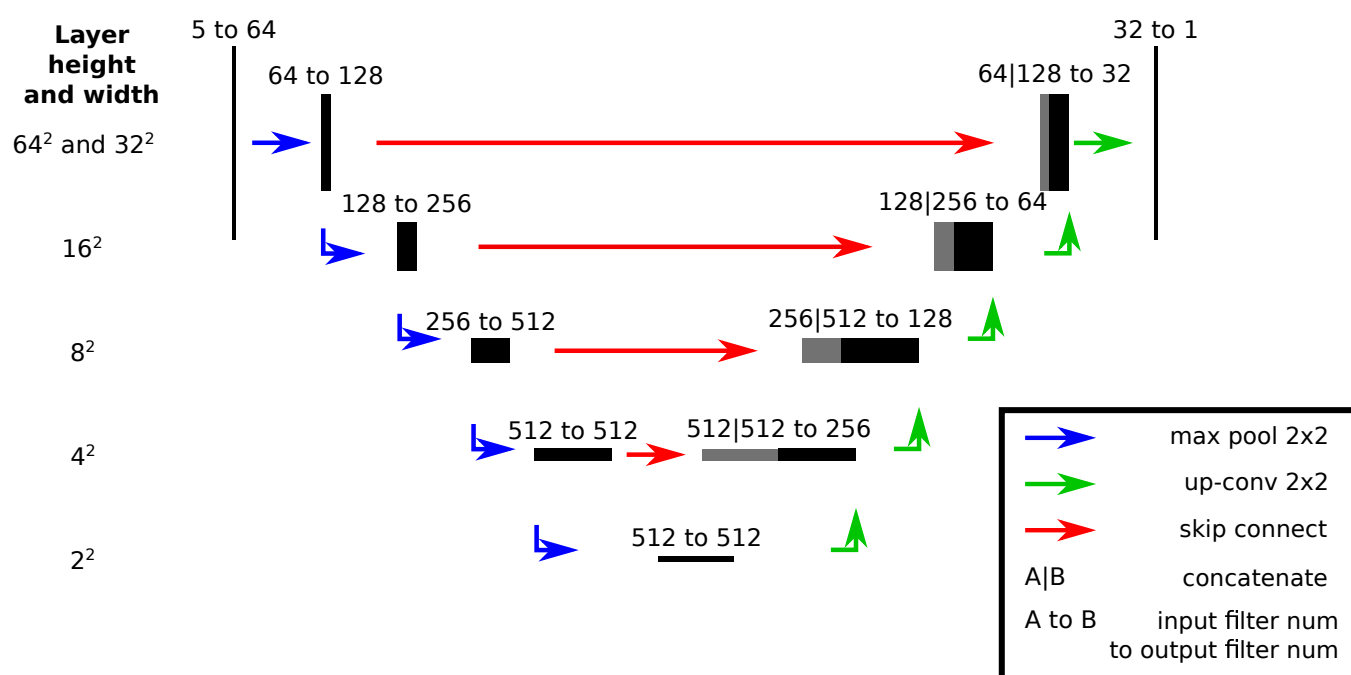


Figure 1. Schematic illustration of the U-net-based model used in this work. The left part shows the encoder reducing the input image size from 64×64 px to 2×2 px. The right part shows the decoder increasing the image size again to 64×64 px.

2.2. Data Source

In our approach, we rely on historical satellite images of pipelines during their construction phase. A broad collection of construction dates for European gas transport pipelines is provided in the open-source database of the non-governmental organization *Global Energy Monitor* [33]. To identify a single consistent image source, we cross-checked construction dates with the operation time of a variety of suitable satellite missions. We then chose the Landsat 5 satellite mission as it provides open-access imagery spanning from 1984 to 2013. The Landsat program is operated by the United States Geological Survey (USGS) and the National Aeronautics and Space Administration (NASA). The onboard mounted whisk broom scanner, named *Thematic Mapper* (TM), supplies suitable satellite imagery for our purposes. The image scenes cover an area of $185 \text{ km} \times 172 \text{ km}$ with a resolution of 120 m px^{-1} for its thermal band and 30 m px^{-1} for the remaining bands. The instrument's resolution is not sufficient to detect gas transport pipelines, which commonly have only diameters of up to 1 m. It is, however, possible to detect construction pathways, so-called *right of ways* (ROWs). During the construction phase, ROWs are cleared from any vegetation alongside a width of 16 to 28 m [34]. We have used satellite imagery depicting ROWs to create a training data set for Great Britain. We did not necessarily choose images with the lowest cloud coverage index, but we have always considered images with the clearest view of the pipeline pathway. The data set was labeled with geo-referenced data for the British gas transmission grid (see Figure 2). This data is openly provided by the gas network operator *National Grid* on their web page [35]. The data we will use for testing was created from images of the NEL (Nordeuropäische Erdgasleitung) pipeline in Northern Germany (see Figure 3) to check how well the model generalizes to an unknown data set. The geo-referenced data of the NEL pipeline was obtained from *OpenStreetMap* (OSM) using the *esy-osmfilter* [7] *Python* package.



Figure 2. Gas pipeline routes for model training and validation: Gas Transport Network Great Britain [35].



Figure 3. Gas pipeline routes for model testing: NEL Pipeline in Northern Germany [36].

2.3. Image Data Processing

In the following, we discuss the technical details of the image labeling process with which we have classified pixels as *pipelines* or as *background*. First, we have created ground truth data with the data of the British pipelines with the help of the *Python* package *Shapely* [37]. In this process, all 6 visual bands of the satellite imagery were normalized to a range between 0 and 1 using the normalization factor 10^{-4} as suggested by the USGS [38]. Pixels with invalid values were replaced with the value 0. Finally, the preprocessed satellite images and corresponding ground truths were cropped into image-mask pairs of the dimensions 64×64 px (see Figure 4). It was hard to visually confirm the pipeline's

construction on images with few pixels tagged as *pipeline*. Thus, only image-mask pairs were kept for training and testing if they contained more than 50 px tagged as *pipeline*.

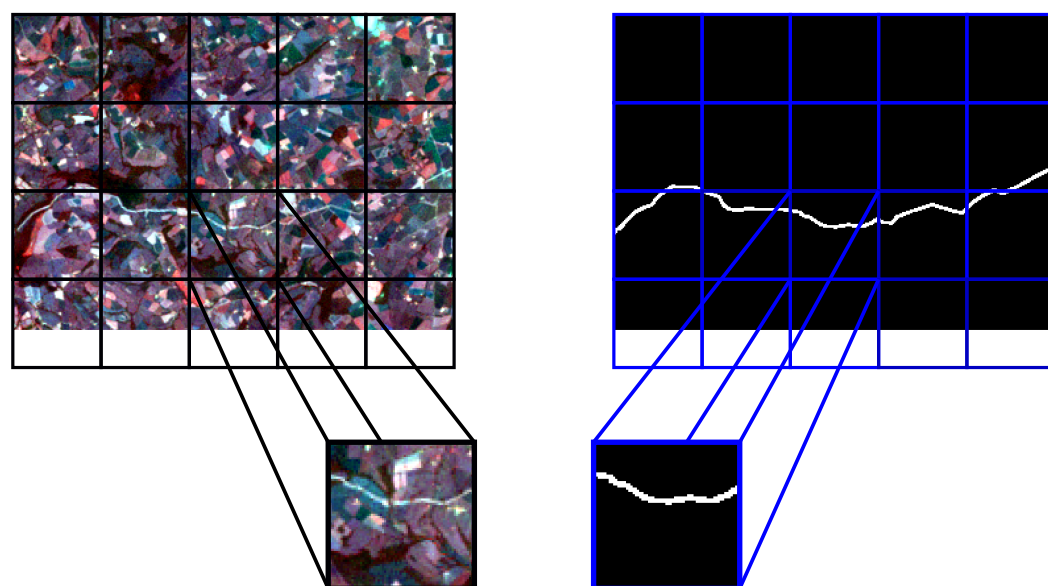


Figure 4. Schematic illustration of the cropping process used to create image-mask pairs. The left side shows the satellite image, and the right side shows the corresponding mask.

During an additional visual inspection of the image-mask pairs, special care was taken to ensure that no significant mislabeling of the images occurred, that the pipelines were already entirely constructed at that time, and that the ROW was clearly visible. This process obtained 324 image-mask pairs for the model training and 33 image-mask pairs for model testing. Due to the limited data on pipeline pathways and construction dates, the overall number of training images is considered small. Therefore, data augmentation by horizontal flip, vertical flip, rotation of 90° , transpose, and 3×3 grid shuffle was used on the image-mask pairs during each training cycle to increase the number of available training data (see Figure 5). Each augmentation was applied with a probability of 50%. The application of multiple augmentations was possible, and the original resolution of 64×64 px is always preserved.

2.4. Training and Validation Process

Like mentioned before, the U-Net architecture is well suited for problems like the detection of pipeline-like objects. However, to make sure it realizes its full potential, the so-called *hyperparameters* need to be tuned. The most crucial hyperparameter according to [39,40] is the *learning rate*. The learning rate defines the step size the optimizer takes towards the minimum of the loss function [41,42]. Our model was trained with initial learning rates of 10^{-3} , 10^{-4} , 10^{-5} , and 10^{-6} using *Dice loss* (DL) as a loss function [43,44]. Dice loss is based on the Dice coefficient (DC), which calculates the similarity between two images, in this case, between the model's prediction and the ground truth (see Figure 6). It ranges between 0 and 1, where 0 indicates no overlap between prediction and ground truth and 1 indicates a complete overlap. Loss functions need to be minimizable; hence, the Dice loss is defined as $DL = 1 - DC$. Further, we applied the Adam optimizer [45] for the gradient-based optimization of the weights using adaptive learning rates. The authors of the Adam optimizer recommend an initial learning rate of 10^{-3} . The remaining parameters of the optimizer were set to the recommended values given in [45].

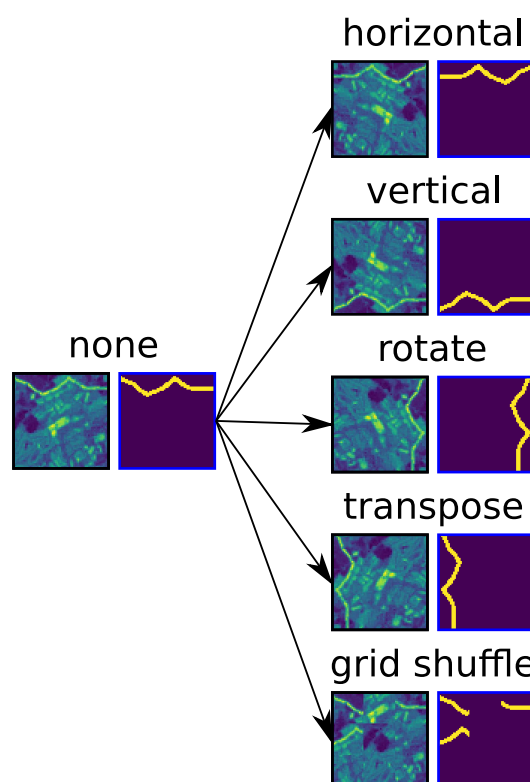


Figure 5. Overview of five augmentations used on the image-mask pairs during the training process.

$$\text{Dice coefficient} = \frac{2 \times \text{area of intersection}}{\text{total area}} = \frac{2 \times \text{[Intersection]}}{\text{[Area 1]} + \text{[Area 2]}}$$

Figure 6. Schematic illustration of the calculation of the Dice coefficient.

We have chosen *Intersection over Union* (IoU) as a scoring metric, which is common for evaluating the performance of image segmentation methods for imbalanced classes. The IoU is a number between 0 and 1, and the higher the number, the better the prediction agrees with the ground truth [46]. Predictions with an IoU score of greater than 0.5 are generally considered as “good” [47]. As we only have a small amount of image-mask data, the validation will result in a relatively noisy estimate of predictive performance. This can be partially compensated by using *k-fold cross-validation* [41]. Thus, we split our data into five folds, where one fold is left out for validation, and the other four are used for model training. This procedure is repeated five times such that each sub-sample is used once for the model validation. This allowed us to train our model for each learning rate with five different validation and training data sets combinations. A complete training process consists of a maximum of up to 200 epochs. An epoch is defined as a complete cycle of the training data set through the entire training process. Like mentioned before, an augmentation was applied to each image during every training cycle with a certain probability. No augmentations were applied to the validation data. The training process consists of assessing the model output by the loss function after each image and updating the model parameters by the optimizer. After each epoch, the model was validated on the validation fold using the IoU metric. The learning rate is reduced by a factor of 0.1 if the validation IoU does not change for 15 epochs. This increases the chance of convergence towards a local optimum at the cost of more iteration steps. The training is stopped if the validation IoU does not improve for 20 epochs.

3. Results and Discussion

In this chapter, we cover the training and testing and the overall performance of our model by discussing the IoU scores, the quality of the ground truths, the visual inspection of the prediction, and the generalization capabilities of the model.

3.1. Training and Validation

The validation IoU and the loss calculated on the validation set are common metrics to guide the training process and measure the overfitting degree. The maximal validation IoU and minimal validation loss of all epochs for each split were averaged for each learning rate. The mean maximal validation IoUs and minimal validation losses for each learning rate can be seen in Figure 7. The training and validation IoU scores of each training epoch of one of the five cross-validation splits are plotted over the number of epochs alongside the mean maximal validation IoU score of the corresponding learning rate in Figure 8a–d. For the learning rates 10^{-3} , 10^{-4} , and 10^{-5} , the mean maximal validation IoU scores and minimal validation losses are very consistent. For the 10^{-6} learning rate, the very low mean maximum validation IoU score and the very high mean minimum validation loss suggest the non-convergence for all splits.

Figure 8a–c confirms this assumption. Further, we notice high fluctuations of the validation IoU score between each training epoch. These fluctuations can be explained by the relatively low number of training samples of 324 image-mask pairs compared to, e.g., the Massachusetts Roads data set containing more than 1000 image-mask pairs 1000×1500 px each [48]. Figure 8d shows more minor fluctuations but no trend of convergence.

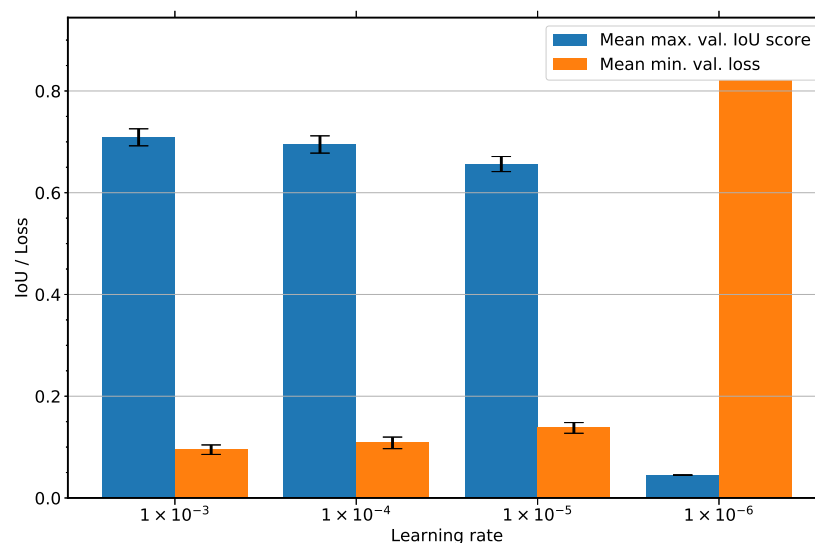


Figure 7. Minimum loss and maximum IoU score averaged over all training splits for each learning rate with standard deviation.

3.2. Testing

We test the ability of our fully or nearly converging models to generalize to another geographic region with the data set of the NEL pipeline in Northern Germany consisting of 33 images. The IoU scores and the losses were averaged for each learning rate (see Figure 9). The highest mean IoU of 0.56 ± 0.04 was measured for the learning rate of 10^{-4} with a mean loss of 0.31 ± 0.03 . The lowest IoU of 0.52 ± 0.03 and highest loss of 0.34 ± 0.02 were obtained for the learning rate of 10^{-5} . The best performing split of the learning rate of 10^{-4} was found to be split 5 with an IoU score of 0.58 and a loss of 0.28. This means it is still performing 0.13 worse in terms of IoU score than the mean maximum validation IoU score during training. To get a better picture of a possible cause, the performance of the trained model for the best performing split (5th split of learning rate of 10^{-4}) was further visually analyzed (see Figure 10).

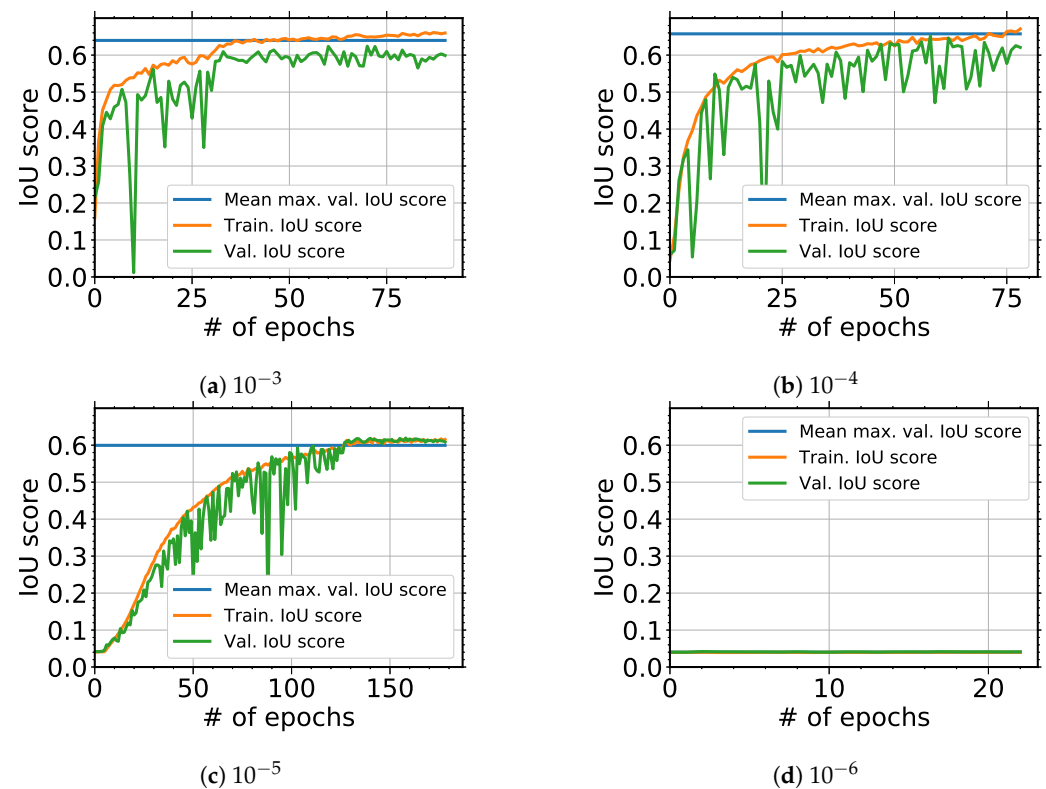


Figure 8. IoU over number of epochs for one split for the different learning rates ranging from 10^{-3} in (a) to 10^{-6} in (d).

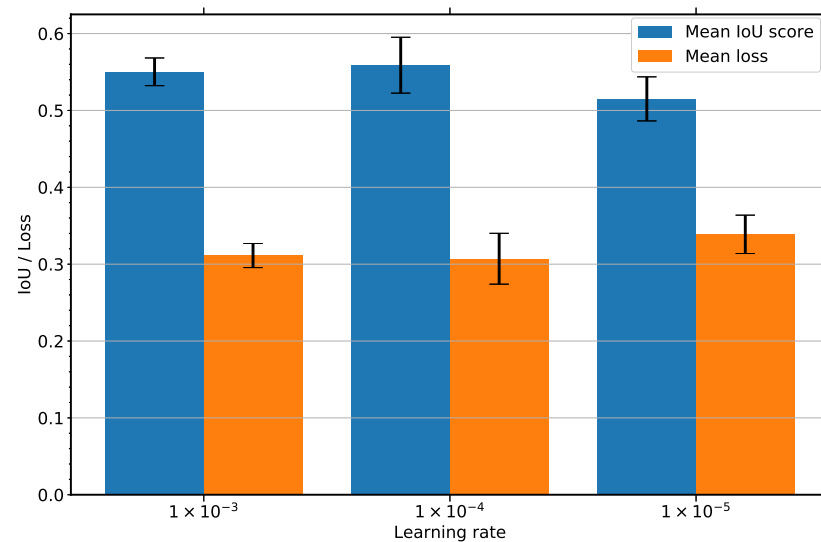


Figure 9. Mean loss and IoU score of trained models applied to the test data set of the NEL pipeline for each learning rate plotted with the standard deviation.

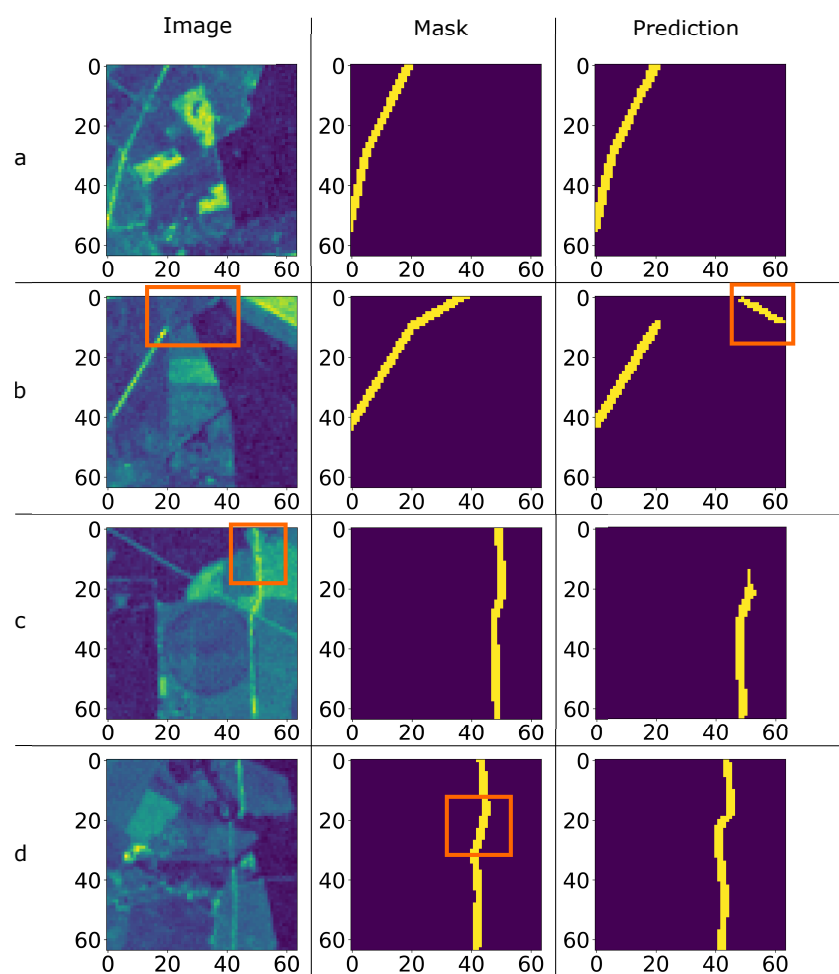


Figure 10. A selection of original satellite images of the NEL pipeline, ground truths, and predictions from the trained model. The orange boxes on the left indicate false negatives, in the middle are the labeling errors, and on the right the false negatives. (a) shows an accurately predicted pipeline, (b,c) show prediction mistakes, and (d) depicts a case of an incorrectly labeled ground truth.

We state that our model can reliably predict pipeline pathways. Figure 10a shows an input image that contains a high number of pixels belonging to infrastructure visually similar to the pipeline pathway. The model is still able to accurately predict the path of the pipeline without any incorrect labeling. However, in Figure 10b, parts of the image that are visually close to the pipeline, e.g., streets, are incorrectly labeled as a *pipeline* (which are false positives), and parts of the pipeline with low contrast to the background are falsely labeled as background (which are false negatives). It is not conclusive why only this small part is labeled as pipeline and not similar areas like in Figure 10a or c. Again, Figure 10c shows the case of false negatives due to visual similarities between the pipeline pathway and the background. However, our model could still correctly predict the proper course of the pipeline even for inaccurately labeled ground truth images (see Figure 10d).

Overall, the visual inspection of the prediction indicates the reliability of the model. The lower IoU score for the test data set can be attributed to multiple factors. The georeferenced data originated from OpenStreetMap and not directly from the responsible TSO. This could have led to additional errors in the labeling process, resulting in more inaccurate ground truths, as those are the basis for assessing the model's performance. Another factor could be the differences in the geographical regions between the training data set and the validation data set. The course of a pipeline appears visually similar throughout Europe. The background, however, is sometimes subject to dramatic changes due to, e.g., the change of the observed vegetation, infrastructure, or soil and crops type. Geographically, Great Britain is located in the “boreal” region, whereas Germany is in the “continental” region.

However, there are even more significant geographical differences between European countries, as is the case, for example, in the “Mediterranean” region, which we have not investigated yet. The model was not trained on the test data set and was therefore unable to learn those features. This may have resulted in more false-positive labels and ultimately in a lower IoU score.

Generally, the quality of the labeling could only be confirmed visually, and the authors had to trust the accuracy of the data provided by OpenStreetMap or the TSO. Occasional labeling errors could be detected visually for the training and testing data set (e.g., Figure 10d). Further, a constant temporal difference between the construction of the pipeline and the recording of the satellite image could not be achieved due to the cloud coverage. Finally, due to the resolution of the images, the pipeline labels were set to a width of 2 pixels. However, the ROW varies, like mentioned above, between 16–28 m which corresponds to instead 1–2 pixels. All the described effects mentioned above may significantly decrease ground truth quality, thus diminishing the overall IoUs.

In summary, Figure 10 indicates the strengths and weaknesses of the model’s ability to predict pipeline pathways on a small scale. In Figure 11 a pipeline section of the NEL pipeline of 22 sub-images (64×64 px) shall demonstrate that the small scale prediction errors play a minor role when it comes to detecting pipelines on a larger scale. The continuous pathway can easily be reconstructed from the prediction depicted in Figure 12. The model performs well on Landsat 5 images of regions with vegetation similar to the areas it was trained with. To extend the model’s geographic limitations, more training data of areas with different vegetation is needed. Additionally, the model is limited to Landsat 5 data. To detect pipelines on images of various sources, the model needs to be retrained with these images. Thus, the current model can only detect pipelines build during the service time of Landsat 5 between 1984 and 2013.



Figure 11. Section of the NEL pipeline composed of 22 sub-images (64×64 px).



Figure 12. Model’s prediction using the 22 sub-images depicted in Figure 11.

4. Summary

We have shown that historic open-source satellite images are suited to detect pipeline pathways automatically. We used satellite images of the Landsat 5 satellite created between 1984 and 2013 to train a deep learning algorithm to provide pixel-level identification of pipeline pathways during their construction phase. The model was trained with imagery of the gas pipeline transport network of Great Britain. We have successfully used training data augmentation and k-fold cross-validation to compensate for the limited amount of available training data of only $324\ 64 \times 64$ px images. The best results for our model were obtained with a learning rate of 10^{-3} converging at a mean max. val. IoU score of 0.71 ± 0.02 . IoU scores of greater than 0.5 are generally considered “good”, and we, therefore, state that the used model can confidently identify pipeline courses. In this work, we have also shown that our model can detect a major pipeline in the geographic region of Northern Germany. The best mean IoU score of 0.56 ± 0.04 was obtained using the model trained with a learning rate of 10^{-4} . This indicates that the model can generalize to other geographic regions outside of the training data set. In general, we can state that the training of a deep learning network to identify the pixel-wise detection of pipeline courses is possible despite low-resolution satellite imagery of only 30 m px^{-1} .

Author Contributions: Conceptualization, A.P. and J.D.; methodology, J.D. and M.Z.; software, J.D.; validation, A.P. and M.Z.; formal analysis, J.D.; investigation, J.D. and A.P.; resources, W.M.; data curation, J.D.; writing—original draft preparation, J.D. and A.P.; writing—review and editing, M.Z. and W.M.; visualization, J.D. and A.P.; supervision, W.M.; project administration, W.M.; funding acquisition, W.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded as part of the DLR Institute for Networked Energy Systems project “SciGRID_gas” by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the funding of the 6. Energieforschungsprogramm der Bundesregierung. Funding Code: 03ET4063.

Data Availability Statement: The data presented in this study are available on request from the corresponding author. The data are not publicly available due to large file sizes.

Acknowledgments: The authors like to acknowledge the contribution of Hendrik-Pieter Tetens, Jan Diettrich, Ontje Lünsdorf, Karl-Kiên Cao, and Susanne Weyand.

Conflicts of Interest: The authors declare no conflict of interest. The funding agency had no role in the design of the study, in the collection, analyses, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

References

1. Brown, T.; Schlachberger, D.; Kies, A.; Schramm, S.; Greiner, M. Synergies of sector coupling and transmission reinforcement in a cost-optimised, highly renewable European energy system. *Energy* **2018**, *160*, 720–739. [\[CrossRef\]](#)
2. Qadrdan, M.; Abeysekera, M.; Chaudry, M.; Wu, J.; Jenkins, N. Role of power-to-gas in an integrated gas and electricity system in Great Britain. *Int. J. Hydrogen Energy* **2015**, *40*, 5763–5775. [\[CrossRef\]](#)
3. Clegg, S.; Mancarella, P. Integrated Modeling and Assessment of the Operational Impact of Power-to-Gas (P2G) on Electrical and Gas Transmission Networks. *IEEE Trans. Sustain. Energy* **2015**, *6*, 1234–1244. [\[CrossRef\]](#)
4. Clegg, S. Storing renewables in the gas network: modelling of power-to-gas seasonal storage flexibility in low-carbon power systems. *Iet Gener. Transm. Distrib.* **2016**, *10*, 566–575. [\[CrossRef\]](#)
5. Medjroubi, W.; Müller, U.P.; Scharf, M.; Matke, C.; Kleinhans, D. Open data in power grid modelling: New approaches towards transparent grid models. *Energy Rep.* **2017**, *3*, 14–21. [\[CrossRef\]](#)
6. Matke, C.; Medjroubi, W.; Kleinhans, D.; Sager, S. Structure analysis of the German transmission network using the open source model SciGRID. In *Advances in Energy System Optimization*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 177–188.
7. Pluta, A.; Lünsdorf, O. esy-osmfilter: A Python Library to Efficiently Extract OpenStreetMap Data. *J. Open Res. Softw.* **2020**, *8*, 19. [\[CrossRef\]](#)
8. Kunz, F.; Kendziorowski, M.; Schill, W.P.; Weibezahn, J.; Zepter, J.; von Hirschhausen, C.R.; Hauser, P.; Zech, M.; Möst, D.; Heidari, S.; et al. *Electricity, Heat, and Gas Sector Data for Modeling the German System*; Technical Report, DIW Data Documentation; EconStor: Berlin, Germany, 2017.
9. Schmidt, M.; Aßmann, D.; Burlacu, R.; Humpola, J.; Joormann, I.; Kanelakis, N.; Koch, T.; Oucherif, D.; Pfetsch, M.E.; Schewe, L.; et al. Gaslib—A library of gas network instances. *Data* **2017**, *2*, 40. [\[CrossRef\]](#)

10. Van der Werff, H.; Van der Meijde, M.; Jansma, F.; Van der Meer, F.; Groothuis, G.J. A spatial-spectral approach for visualization of vegetation stress resulting from pipeline leakage. *Sensors* **2008**, *8*, 3733–3743. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Zakharov, I.; Adlakha, P.; Puestow, T.; Power, D.; Warren, S.; Howell, M. Monitoring Pipeline Right of Way Using Optical Satellite Imagery. In Proceedings of the 11th Pipeline Technology Conference, Berlin, Germany, 23–25 May 2016.
12. Tran, L.A.; Le, M.H. Robust U-Net-based road lane markings detection for autonomous driving. In Proceedings of the 2019 International Conference on System Science and Engineering (ICSSE), Dong Hoi City, Vietnam, 20–21 July 2019; IEEE: New York, NY, USA, 2019; pp. 62–66.
13. Wei, Y.; Zhang, K.; Ji, S. Road Network Extraction from Satellite Images Using CNN Based Segmentation and Tracing. In Proceedings of the IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; IEEE: New York, NY, USA, 2019; pp. 3923–3926.
14. Dasenbrock, J. Pipeline Detection with Satellite Images Using Machine Learning. Master's Thesis, University of Oldenburg, Oldenburg, Germany, 2020.
15. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [\[CrossRef\]](#)
16. Yuan, X.; Ou, C.; Wang, Y.; Yang, C.; Gui, W. A layer-wise data augmentation strategy for deep learning networks and its soft sensor application in an industrial hydrocracking process. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *32*, 3296–3305. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Yuan, X.; Li, L.; Shardt, Y.A.; Wang, Y.; Yang, C. Deep learning with spatiotemporal attention-based LSTM for industrial soft sensor model development. *IEEE Trans. Ind. Electron.* **2020**, *68*, 4404–4414. [\[CrossRef\]](#)
18. Ching, T.; Himmelstein, D.S.; Beaulieu-Jones, B.K.; Kalinin, A.A.; Do, B.T.; Way, G.P.; Ferrero, E.; Agapow, P.M.; Zietz, M.; Hoffman, M.M.; et al. Opportunities and obstacles for deep learning in biology and medicine. *J. R. Soc. Interface* **2018**, *15*, 20170387. [\[CrossRef\]](#)
19. Wurm, M.; Droin, A.; Stark, T.; Geiß, C.; Sulzer, W.; Taubenböck, H. Deep learning-based generation of building stock data from remote sensing for urban heat demand modeling. *Isprs Int. J. -Geo-Inf.* **2021**, *10*, 23. [\[CrossRef\]](#)
20. Knopp, L.; Wieland, M.; Rättich, M.; Martinis, S. A deep learning approach for burned area segmentation with Sentinel-2 data. *Remote Sens.* **2020**, *12*, 2422. [\[CrossRef\]](#)
21. Voulodimos, A.; Doulamis, N.; Doulamis, A.; Protopapadakis, E. Deep learning for computer vision: A brief review. *Comput. Intell. Neurosci.* **2018**, *2018*, 8349
22. Guo, Y.; Liu, Y.; Georgiou, T.; Lew, M.S. A review of semantic segmentation using deep neural networks. *Int. J. Multimed. Inf. Retr.* **2018**, *7*, 87–93. [\[CrossRef\]](#)
23. University of Freiburg. Our U-Net Wins Two Challenges at ISBI 2015. Available online: <https://lmb.informatik.uni-freiburg.de/people/ronneber/isbi2015/> (accessed on 12 October 2020).
24. Goceri, E. Challenges and recent solutions for image segmentation in the era of deep learning. In Proceedings of the 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), Ataşehir/İstanbul, Turkey, 7–9 November 2019; IEEE: New York, NY, USA, 2019; pp. 1–6.
25. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
26. Zech, M.; Ranalli, J. Predicting PV Areas in Aerial Images with Deep Learning. In Proceedings of the 2020 47th IEEE Photovoltaic Specialists Conference (PVSC), Online, 15 June–21 August 2020; IEEE: New York, NY, USA, 2020; pp. 767–774.
27. Çiçek, Ö.; Abdulkadir, A.; Lienkamp, S.S.; Brox, T.; Ronneberger, O. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Athens, Greece, 17–21 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 424–432.
28. Islam, M.; Vibashan, V.S.; Jose, V.J.M.; Wijethilake, N.; Utkarsh, U.; Ren, H. Brain Tumor Segmentation and Survival Prediction Using 3D Attention UNet. In *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*; Crimi, A.; Bakas, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 262–272.
29. Frajberg, D.; Fraternali, P.; Torres, R.N. Convolutional neural network for pixel-wise skyline detection. In Proceedings of the International Conference on Artificial Neural Networks, Alghero, Italy, 11–14 September 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 12–20.
30. Bai, Y.; Mas, E.; Koshimura, S. Towards operational satellite-based damage-mapping using u-net convolutional network: A case study of 2011 tohoku earthquake-tsunami. *Remote Sens.* **2018**, *10*, 1626. [\[CrossRef\]](#)
31. Zhang, Z.; Liu, Q.; Wang, Y. Road extraction by deep residual u-net. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 749–753. [\[CrossRef\]](#)
32. Yakubovskiy, P. Segmentation Models. Available online: https://github.com/qubvel/segmentation_models (accessed on 20 November 2019).
33. Global Energy Monitor. National Transmission System. Available online: https://www.gem.wiki/National_Transmission_System (accessed on 16 September 2020).
34. Kennedy, J.L. *Oil and Gas Pipeline Fundamentals*; Pennwell Corp.: Tulsa, OK, USA, 1984.
35. National Grid UK. Network Route Maps. Available online: <https://www.nationalgrid.com/uk/gas-transmission/land-and-assets/network-route-maps> (accessed on 10 September 2020).

36. Europepipe. Referenzprojekte. Available online: <https://www.europipe.com/de/referenzen/referenzprojekte#c1538> (accessed on 3 February 2020).
37. Gillies, S. Shapely: Manipulation and Analysis of Geometric Objects. Available online: <https://github.com/Toblerity/Shapely> (accessed on 10 May 2020).
38. Sayler, K.; Zanter, K. *Landsat 4-7 Collection 1 (C1) Surface Reflectance (LEDAPS) Product Guide*; USGS: Reston, VA, USA, 2020.
39. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *28*, 2222–2232. [CrossRef] [PubMed]
40. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016; Chapter 11. Available online: <http://www.deeplearningbook.org> (accessed on 8 August 2021).
41. Bishop, C.M. *Pattern Recognition and Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2006; Chapter 1.
42. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
43. Sudre, C.H.; Li, W.; Vercauteren, T.; Ourselin, S.; Cardoso, M.J. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 240–248.
44. Jadon, S. A survey of loss functions for semantic segmentation. In Proceedings of the 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Online, 27–29 October 2020; IEEE: New York, NY, USA, 2020; pp. 1–7.
45. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
46. Ulku, I.; Akagunduz, E. A survey on deep learning-based architectures for semantic segmentation on 2d images. *arXiv* **2019**, arXiv:1912.10230.
47. Géron, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow: Concepts, Tools, and Techniques to Build Intelligent Systems*; O'Reilly Media: Sebastopol, CA, USA, 2019; Chapter 4.
48. Mnih, V. Machine Learning for Aerial Image Labeling. Ph.D. Thesis, University of Toronto, Toronto, ON, Canada, 2013.